# EXPERIMENT - 23

## Object Oriented Programming Lab

### Aim

Write a program to overload new and delete operator.

Syeda Reeha Quasar
14114802719
4C7

# EXPERIMENT – 23

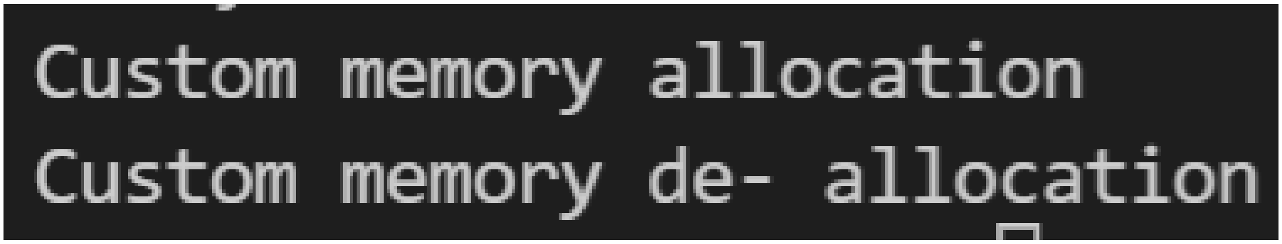
## Aim:

Write a program to overload new and delete operator.

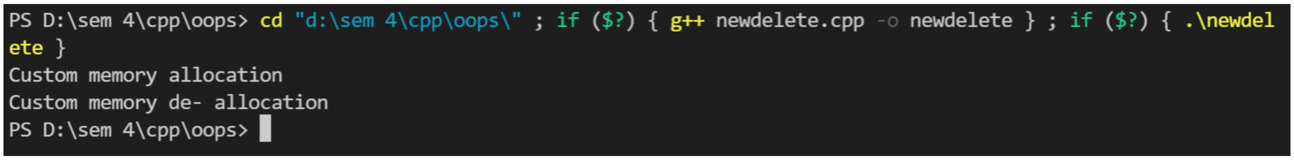
## Source Code:

```
#include <iostream>
using namespace std;

class CustomMemory
{

public:
    void *operator new(size_t objectSize); //Overloaded new
    void operator delete(void *ptr);       //Overloaded delete
};

void *CustomMemory::operator new(size_t objectSize)
{
    cout << "Custom memory allocation" << endl;
    return malloc(objectSize);
}
void CustomMemory::operator delete(void *ptr)
{
    cout << "Custom memory de- allocation" << endl;
    free(ptr);
}

int main()
{

    // call overloaded new from the class
    CustomMemory *obj = new CustomMemory();

    // call overloaded delete
    delete obj;
}
```

## Output:



```
Custom memory allocation
Custom memory de- allocation
```



```
PS D:\sem 4\cpp\oops> cd "d:\sem 4\cpp\oops\" ; if ($?) { g++ newdelete.cpp -o newdelete } ; if ($?) { .\newdel
ete }
Custom memory allocation
Custom memory de- allocation
PS D:\sem 4\cpp\oops> 
```

# Viva Questions

## Q1) What are causes of function overloading?

Ans.

When the compiler is unable to decide which function is to be invoked among the overloaded function, this situation is known as **function overloading**.

When the compiler shows the ambiguity error, the compiler does not run the program.

**Causes of Function Overloading:**

- Type Conversion.
- Function with default arguments.
- Function with pass by reference.

## *Q2) What is overloading?*

Ans.

C++ allows you to specify more than one definition for a **function** name or an **operator** in the same scope, which is called **function overloading** and **operator overloading** respectively.

An overloaded declaration is a declaration that is declared with the same name as a previously declared declaration in the same scope, except that both declarations have different arguments and obviously different definition (implementation).

*When you call an overloaded **function** or **operator**, the compiler determines the most appropriate definition to use, by comparing the argument types you have used to call the function or operator with the parameter types specified in the definitions. The process of selecting the most appropriate overloaded function or operator is called **overload resolution**.*

## Q3) What is purpose of operator overloading?

Ans.

The **purpose of operator overloading** is to provide a special meaning of an **operator** for a user-defined data type. With the help of **operator overloading**, you can redefine the majority of the C++ **operators**. You can also **use operator overloading** to perform different operations using one **operator**.